Remarks

In the present response, no claims are amended. Claims 1-51 are presented for examination.

I. Objection to Drawings

The Office Action objects to the drawings "because the specification refers to intermediate modified computer program 124, however 124 is an Annotation Interference System." Applicants respectfully traverse.

FIG. 1 correctly illustrates Annotation Inference System 130 and Intermediate Modified Computer Program 124. The specification is consistent with this nomenclature. For example, the specification at p. 9, lines 7 and 13 discusses the annotation inference system 130; and the specification at p. 20, lines 9-10 discusses the intermediate modified computer program 124.

Withdrawal of this objection is respectfully requested.

II. Claims Rejection: 35 USC § 103

Claims 1-8, 14-28, 32-42, and 46-51 are rejected under 35 USC § 103(a) as being unpatentable over USPN 5,423,027 (hereafter Jackson) in view of USPN 6,343,376 (hereafter Saxe). Applicants respectfully traverse.

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art cited must teach or suggest all the claim limitations. *See* M.P.E.P. § 2143. Applicants assert that the rejection does not satisfy any of the criteria of MPEP § 2143. Examples are provided below.

Claim 1

Claim 1 recites numerous limitations that are not taught or suggested in Jackson. For example, claim 1 recites applying a program checking tool to produce one or more warnings. The claim then recites mapping at least one of the warnings into at least one

10

annotation modification. Jackson does not teach or suggest mapping a **warning** into an annotation modification. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67). "The checker program reviews the program in light of the aspect specifications and produces output which identifies errors in the program" (col. 6, lines 1-3).

Thus, Jackson teaches adding lines of aspect specification to the code and then reviewing the code to produce output that identifies errors in the program. Nowhere docs Jackson teach or suggest that the checker program produces a warning and then **maps the warning into an annotation modification.** In Jackson, the checker programs reviews the program and then "produces output which identifies errors in the program." Jackson does not review the program and then map the warnings into an annotation modification.

As another example, claim 1 recites modifying the computer program in accordance with the at least one annotation modification. Jackson does not teach or suggest this limitation. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67). "The checker program reviews the program in light of the aspect specifications and produces output which identifies errors in the program" (col. 6, lines 1-3).

Thus, Jackson teaches that a programmer or the checker program can annotate the code with lines of aspect specification. Jackson, however, does not teach or suggest that the checker program annotates the code in accordance with at least one annotation modification. Instead, Jackson merely states that the code can be annotated. This teaching in Jackson should be compared with the recitation in claim 1: "modifying the computer program **in accordance with said at least one annotation modification**" (emphasis added).

As yet another example, claim 1 recites:

> d) repeating each of steps a), b) and c) until no warnings produced in step a) are suitable for mapping into an annotation modification.

11

Serial No.: 10/005,923
Response to OA of 02/10/2005

The Office Action admits that "Jackson does not expressly disclose repeating each of steps a, b and c until no warnings produced" (see OA p. 4). Applicants agree with this admission. The Office Action, however, attempts to cure this deficiency with Saxe. Applicants respectfully disagree.

The Office Action cites Saxe at col. 7, lines 19-26 for teaching the limitation of d) repeating each of steps a, b, and c. This section of Saxe is reproduced below for convenience:

> In an extension of the present invention, detection by the
> prover 204 of an error or inconsistency in the formula being
> analyzed permits the output of the prover to be used as the basis
> for assisted amelioration or elimination of the diagnosed formal
> error, as shown at 209. The edited source code may optionally be
> fed back to the verification condition generator 202, causing the
> process to repeat from the beginning as described earlier.

This section of Saxe merely teaches that a section of source code can be fed back into a verification condition generator. By contrast, claim 1 recites repeating each of three steps: (a) applying a checking tool to produce one or more warnings, (b) mapping at least one warning into an annotation modification, and (c) modifying the computer program in accordance with the annotation modification. Nowhere does Saxe teach or suggest that steps a, b, and c of claim 1 are repeated until no warnings are produced. Instead, Saxe merely teaches that a section of source code can be fed back into a verification generator.

For at least these reasons, claim 1 and its dependent claims are allowable over Jackson in view of Saxe.

Claim 22

Claim 22 recites numerous limitations that are not taught or suggested in Jackson. For example, claim 22 recites mapping at least one warning into at least one annotation modification. Applicants reiterate the arguments presented in connection with claim 1. Namely, Jackson does not teach or suggest mapping a **warning** into an annotation

12

modification. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67).

As another example, claim 22 recites updating the computer program in accordance with the annotation modification. Applicants reiterate the arguments presented in connection with claim 1. Namely, Jackson teaches that a programmer or the checker program can annotate the code with lines of aspect specification. Jackson, however, does not teach or suggest that the checker program updates the computer program in accordance with the annotation modification.

As yet another example, claim 22 recites repeatedly invoking the program checking tool, warning mapper, and program updater until no warnings are produced by the program checking tool. Applicants reiterate the arguments presented in connection with claim 1. Namely, Saxe merely teaches that a section of source code can be fed back into a verification condition generator.

For at least these reasons, claim 22 and its dependent claims are allowable over Jackson in view of Saxe.

Claim 36

Claim 36 recites numerous limitations that are not taught or suggested in Jackson. For example, claim 36 recites a processor configured to map at least one warning into at least one annotation modification. Applicants reiterate the arguments presented in connection with claim 1. Namely, Jackson does not teach or suggest mapping a **warning** into an annotation modification. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67).

As another example, claim 36 recites modifying the computer program in accordance with the annotation modification. Applicants reiterate the arguments presented in connection with claim 1. Namely, Jackson teaches that a programmer or the checker program can annotate the code with lines of aspect specification. Jackson, however, does not teach or suggest that the checker program modifies the computer program in accordance with the annotation modification.

13

Serial No.: 10/005,923
Response to OA of 02/10/2005

As yet another example, claim 36 recites a processor configured to repeat applying the program checking tool, mapping the warnings, and modifying the program until no warnings are produced by the program checking tool. Applicants reiterate the arguments presented in connection with claim 1. Namely, Saxe merely teaches that a section of source code can be fed back into a verification condition generator.

For at least these reasons, claim 36 and its dependent claims are allowable over Jackson in view of Saxe.

Claim 50

Claim 50 recites numerous limitations that are not taught or suggested in Jackson. For example, claim 50 recites mapping at least one warning into at least one annotation modification. Applicants reiterate the arguments presented in connection with claim 1. Namely, Jackson does not teach or suggest mapping a **warning** into an annotation modification. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67).

As yet another example, claim 50 recites repeating each of a, b, and c until no new warnings are produced Applicants reiterate the arguments presented in connection with claim 1. Namely, Saxe merely teaches that a section of source code can be fed back into a verification condition generator.

For at least these reasons, claim 50 is allowable over Jackson in view of Saxe.

Claim 51

Claim 51 recites numerous limitations that are not taught or suggested in Jackson. For example, claim 51 recites applying a program check tool to produce one or more warnings about inconsistencies between the computer program and one or more annotations. Jackson does not teach or suggest producing warnings about inconsistencies between the computer program and one or more annotations. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67).

As another example, claim 51 recites mapping at least one warning into at least one annotation modification. Applicants reiterate the arguments presented in connection

14

with claim 1. Namely, Jackson does not teach or suggest mapping a **warning** into an annotation modification. By contrast, Jackson merely teaches that a programmer or checker program can add lines of aspect specification to the code (see col. 5, lines 63-67)

As yet another example, claim 51 recites removing from the computer program an annotation that is mentioned by at least one warning. Neither Jackson nor Saxe teach or suggest such a limitation. For example, nowhere does Jackson discuss removing from a program an annotation that is mentioned in a warning. Further, the Office Action has not pointed to a location in either Jackson or Saxe that teaches or suggests this recitation.

As yet another example, claim 51 recites repeating steps b, c, and d until no new warnings are produced in b. Applicants reiterate the arguments presented in connection with claim 1. Namely, Saxe merely teaches that a section of source code can be fed back into a verification condition generator.

For at least these reasons, claim 51 is allowable over Jackson in view of Saxe.

III. Claims Rejection: 35 USC § 103

Claim 9 is rejected under 35 USC § 103(a) as being unpatentable over Jackson in view of Saxe and further in view of USPN 6,553,362 (hereafter Saxe 362). Applicants respectfully traverse.

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art cited must teach or suggest all the claim limitations. *See* M.P.E.P. § 2143. Applicants assert that the rejection does not satisfy any of the criteria of MPEP § 2143.

Claim 9 depends from claim 1. As noted above in connection with section II, Jackson in view of Saxe does not teach or suggest all the limitations of claim 1. Saxe 362 fails to cure the deficiencies noted above. Thus, for at least the reasons given in connection with claim 1, claim 9 is allowable.

15

IV. Claims Rejection: 35 USC § 103

Claims 10-13, 29-31, and 43-45 are rejected under 35 USC § 103(a) as being unpatentable over Jackson in view of Saxe and further in view of USPN 6,154,876 (hereafter Haley). Applicants respectfully traverse.

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art cited must teach or suggest all the claim limitations. *See* M.P.E.P. § 2143. Applicants assert that the rejection does not satisfy any of the criteria of MPEP § 2143.

Claims 10-13 depend from claim 1; claims 29-31 depend from claim 22; and claims 43-45 depend from claim 36. As noted above in connection with section II, Jackson in view of Saxe does not teach or suggest all the limitations of claims 1, 22, and 36. Haley fails to cure the deficiencies noted above. Thus, for at least the reasons given in connection with claims 1, 22, and 36, claims 10-13, 29-31, and 43-45 are allowable.

Serial No.: 10/005,923
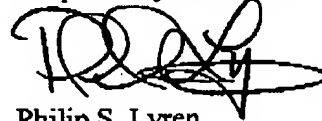Response to OA of 02/10/2005

CONCLUSION

In view of the above, Applicants believe all pending claims are in condition for allowance. Allowance of these claims is respectfully requested.

Any inquiry regarding this Amendment and Response should be directed to Philip S. Lyren at Telephone No. (281) 514-8236, Facsimile No. (281) 514-8332. In addition, all correspondence should continue to be directed to the following address:

Hewlett-Packard Company
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Respectfully submitted,

Philip S. Lyren
Reg. No. 40,709
Ph: 281-514-8236

17